A reflection on the implicitly restarted Arnoldi method for computing eigenvalues near a vertical line

Karl Meerbergen en Raf Vandebril

Karl Meerbergen
Department of Computer Science
KU Leuven, Belgium
Karl.Meerbergen@cs.kuleuven.be

Raf Vandebril
Department of Computer Science
KU Leuven, Belgium
Raf.Vandebril@cs.kuleuven.be

Abstract

In this article, we will study the link between a method for computing eigenvalues closest to the imaginary axis and the implicitly restarted Arnoldi method. The extension to computing eigenvalues closest to a vertical line is straightforward, by incorporating a shift. Without loss of generality we will restrict ourselves here to computing eigenvalues closest to the imaginary axis.

Article information

- Meerbergen, Karl; Vandebril, Raf. *A reflection on the implicitly restarted Arnoldi method for computing eigenvalues near a vertical line*, Linear Algebra and Its Applications, volume 436, issue 8, pages 2828-2844, 2012
- The content of this article is identical to the content of the published paper, but without the final typesetting by the publisher.
- Journal's homepage: http://www.journals.elsevier.com/linear-algebra-and-its-applications/
- Published version: http://dx.doi.org/10.1016/j.laa.2011.07.020
- KU Leuven's repository url: https://lirias.kuleuven.be/handle/123456789/280651

A reflection on the implicitly restarted Arnoldi method for computing eigenvalues near a vertical line[☆]

Karl Meerbergen

Dept. Computer Science, K.U.Leuven, Celestijnenlaan 200A, 3000 Heverlee (Leuven), Belgium

Raf Vandebril

Dept. Computer Science, K.U.Leuven, Celestijnenlaan 200A, 3000 Heverlee (Leuven), Belgium

Abstract

In this article, we will study the link between a method for computing eigenvalues closest to the imaginary axis and the implicitly restarted Arnoldi method. The extension to computing eigenvalues closest to a vertical line is straightforward, by incorporating a shift. Without loss of generality we will restrict ourselves here to computing eigenvalues closest to the imaginary axis.

In a recent publication, Meerbergen and Spence discussed a new approach for detecting purely imaginary eigenvalues corresponding to Hopf bifurcations, which is of interest for the stability of dynamical systems. The novel method is based on inverse iteration (inverse power method) applied on a Lyapunov-like eigenvalue problem. To reduce the computational overhead significantly a projection was added.

This method can also be used for computing eigenvalues of a matrix pencil near a vertical line in the complex plane. We will prove in this paper that the combination of inverse iteration with the projection step is equivalent to Sorensen's implicitly restarted Arnoldi method utilizing well-chosen shifts.

Keywords: Lyapunov eigenvalue problem, Kronecker eigenvalue problem, eigenvalues closest to the imaginary axis, implicitly restarted Arnoldi

1. Introduction

In this article, we will study a method for computing eigenvalues of a large sparse generalized eigenvalue problem, closest to the imaginary axis. This problem is of interest, e.g., for the study of stability of dynamical systems where one is interested in computing Hopf bifurcations. Computing the specific values for which Hopf bifurcations arise, results in large, sparse eigenvalue problems. From earlier work [1–4], we know that detecting eigenvalues near the imaginary axis is not always an easy task. The reason is that many eigenvalue solvers converge to dominant eigenvalues (i.e. eigenvalues of largest absolute magnitude); based on shift-invert strategies, this corresponds to the eigenvalues closest to a target point, called the shift. The shift-and-invert Arnoldi method is such a solver, targeting one specific location in the complex plane rather than the imaginary axis.

The problem we want to solve, can be formulated as computing α and β with $|\alpha|$ as small as possible so that $\lambda = \alpha + \imath\beta$ (with $\imath^2 = -1$) is an eigenvalue of

$$A\mathbf{x} = \lambda B\mathbf{x}. \tag{1}$$

[☆]The second author has a grant as "Postdoctoraal Onderzoeker" from the Fund for Scientific Research–Flanders (Belgium). This paper presents research results of the Belgian Network DYSCO (Dynamical Systems, Control, and Optimization), funded by the Interuniversity Attraction Poles Programme, initiated by the Belgian State Science Policy Office, and by the Research Council K.U.Leuven grants PFV/10/002, OT/10/038 and OT/11/055.

Email addresses: karl.meerbergen@cs.kuleuven.be (Karl Meerbergen), raf.vandebril@cs.kuleuven.be (Raf Vandebril)

URL: http://people.cs.kuleuven.be/karl.meerbergen (Karl Meerbergen), http://people.cs.kuleuven.be/raf.vandebril (Raf Vandebril)

By eliminating β, we find the Kronecker eigenvalue problem

$$\frac{1}{2}(A \otimes B + B \otimes A)\mathbf{z} = \gamma(B \otimes B)\mathbf{z} \tag{2}$$

with $\mathbf{z} = \mathbf{x} \otimes \bar{\mathbf{x}}$. The desired α's are eigenvalues of (2). The advantage of the Kronecker problem is that β is eliminated, so that aiming for the smallest α (actually, the smallest γ) can simply be performed by inverse iteration (also called the inverse power method) applied to (2). Note that, as we shall see in Section 2, not all eigenvalues of (2) lead to eigenvalue pairs of (1). It is thus possible that the γ nearest zero does not correspond to the α nearest zero. Therefore, we compute several γ's by means of inverse subspace iteration, in order to reduce the risk of missing the smallest $|\alpha|$'s. The goal of this paper is to tune the inverse iteration method to the special structure of (2). This will lead to an interpretation of the implicitly restarted Arnoldi method (IRAM) by Sorensen [5–8] applied to the shift-and-invert transformation.

In a recent article [9], a method was presented for computing eigenvalues of the two-parameter eigenvalue problem $(A - \alpha B)\mathbf{x} = \imath\beta M\mathbf{x}$, with A, B and M real. Here, α is a parameter and $\imath\beta$ the eigenvalue. The desired α is the one closest to zero corresponding to a pair of purely imaginary eigenvalues $\pm\imath\beta$. The elimination of β similarly led to a Kronecker eigenvalue problem of dimension $n^2 \times n^2$. Inverse iteration was, however, not executed on the Kronecker eigenvalue problem, but on the corresponding Lyapunov eigenvalue problem. Inverse iteration on the Lyapunov eigenvalue problem requires the solution of a Lyapunov equation. This permits one to exploit low rank structure in the eigenvectors and thereby keeps the computational complexity under control. The dimension of the resulting Lyapunov eigenvalue problem is reduced even more via projection on the subspace generated by the Lyapunov solver and then an extra projection step is needed for storing the solution eigenvector (sometimes referred to as eigenmatrix) of n^2 parameters by $\mathrm{O}(n)$ parameters. The application of this method to (2), i.e., with $M = B$, will lead to a connection with IRAM when a Krylov method is used as Lyapunov solver. Note that there are better methods than Krylov solvers for Lyapunov equations, but the use of Krylov methods is key for the connection with IRAM. We will show that we can interpret IRAM as a subspace iteration method applied to Ritz vectors, obtained by a projection step, where the subspace iteration steps consists of an inexact Lyapunov solver. It should be noted that a connection with subspace iteration was derived in other contexts [10, 11].

The results provided here are a first step towards a better understanding and a more general theoretical framework for studying the approach of [9] and will help us in the development and analysis of alternative methods for specific two-parameter eigenvalue problems. Without the extra projection step for keeping the memory cost low, the method coincides with performing (inexact) inverse iteration on the large Kronecker product problem, which is easily understood. The additional projection step complicates a theoretical study of the convergence behavior. The paper gives an interpretation for the case $M = B$. A minor contribution, is that the paper extends [9] from a single vector iteration to subspace iteration [12, 13].

We assume that A is non-singular. If A would be singular, the eigenvalue nearest the imaginary axis is simply zero. Under mild conditions on A, an eigenvector is found in one iteration of shift-and-invert Arnoldi. We can, of course, replace A by $A - \sigma B$ where σ is a real shift, chosen so that $A - \sigma B$ is non-singular. The extension to eigenvalues closest to the vertical line $\{\lambda : \mathrm{Re}(\lambda) = \sigma\}$ is straightforward by shifting the original problem; therefore, and without loss of generality we will restrict ourselves to the imaginary axis.

The article is organized as follows. In Section 2 we will discuss the related Kronecker and Lyapunov eigenvalue problems. Section 3 discusses how to compute the desired eigenvalues based on inverse subspace iteration on the Lyapunov problem, all projection steps and the preservation of the structure of the desired eigenmatrix are discussed. In Section 4 the link between inverse subspace iteration and implicitly restarted Arnoldi is clarified, it will be shown that the extra projection step corresponds to diminishing the size of the Krylov subspace by executing QR-steps. Section 5 provides some numerical experiments illustrating the applicability of the method. Concluding remarks are given in Section 6.

2. The problem setting and equivalent eigenvalue problems

Computing the eigenvalues closest to the imaginary axis is based on a transformation of the original eigenvalue problem to an equivalent Kronecker and Lyapunov eigenvalue problem. This section discusses these closely related eigenvalue problems.

Consider the generalized eigenvalue problem

$$A\mathbf{x} = \lambda B\mathbf{x}, \tag{3}$$

with $A, B \in \mathbb{R}^{n\times n}$ and nonsingular, whose eigenvalues $\lambda_i = \alpha_i + \imath\beta_i$ $(1 \le i \le n)$ closest to the imaginary axis are desired. Hence, among all λ_i we are interested in the ones with the smallest $|\alpha_i|$. Generically, inverse (subspace) iteration on (3) converges to the smallest $|\lambda_i|$ in absolute value. Therefore, convergence to the smallest $|\alpha_i|$ cannot be guaranteed.

2.1. Kronecker eigenvalue problem

Transforming the generalized eigenvalue problem to a Kronecker eigenvalue problem, eliminating thereby β_i will enable us to overcome this obstacle. The eigenvalue problem $A\mathbf{x} = (\alpha + \imath\beta)B\mathbf{x}$ can be considered as a two-parameter eigenvalue problem, where α and β are now two unknown eigenvalue parameters. Hence, we are interested in the smallest $|\alpha_i|$ which is either real ($\beta_i = 0$) or corresponds to a couple of complex conjugate eigenvalues $\alpha_i \pm \imath\beta_i$. The two-parameter eigenvalue problem has the following relation with the Kronecker eigenvalue problem. (More detailed information on the Kronecker eigenvalue problem, based on the bi-alternate product, can, e.g., be found in [14].)

Theorem 1. *Take $A, B \in \mathbb{R}^{n\times n}$, consider the following two eigenvalue problems:*

$$A\mathbf{x} = \lambda B\mathbf{x}, \tag{4}$$

$$\frac{1}{2}(A\otimes B + B\otimes A)\mathbf{z} = \gamma(B\otimes B)\mathbf{z}. \tag{5}$$

For each real eigenvalue pair $(\lambda, \mathbf{x})$ of Equation (4)*, $\gamma = \lambda$ is an eigenvalue of* (5) *with eigenvector $\mathbf{z} = \mathbf{x}\otimes\mathbf{x}$. For each complex conjugate eigenvalue couple λ and $\bar{\lambda}$ ($\lambda = \alpha + \imath\beta, \beta \neq 0$) of Equation* (4)*, $\gamma = \alpha$ is a double eigenvalue of* (5) *with $\mathbf{z} = \mathbf{x}\otimes\bar{\mathbf{x}}$ and $\bar{\mathbf{z}}$ as eigenvectors.*

Conversely, if γ is an eigenvalue of (5)*, then there are eigenvalues λ_1 and λ_2 from* (4)*, with $2\gamma = \lambda_1 + \lambda_2$. Moreover, $\mathbf{z}$ is a linear combination of $\mathbf{x}\otimes\mathbf{y}$ and $\mathbf{y}\otimes\mathbf{x}$, where $A\mathbf{x} = \lambda_1 B\mathbf{x}$ and $A\mathbf{y} = \lambda_2 B\mathbf{y}$.*

Proof. Due to the appealing nature of the proof, we reconsider some parts of it (a more general form can be found in [9]). We first prove the case of two complex conjugate eigenvalues. Since A, B are real, all complex eigenvalues appear in couples. Consider the eigenpairs $(\alpha + \imath\beta, \mathbf{x})$ and $(\alpha - \imath\beta, \bar{\mathbf{x}})$, we have

$$A\mathbf{x} = (\alpha + \imath\beta)B\mathbf{x} \quad \text{and} \quad A\bar{\mathbf{x}} = (\alpha - \imath\beta)B\bar{\mathbf{x}}.$$

Rewriting these equalities, separating thereby α and β gives us

$$(A - \alpha B)\mathbf{x} = \imath\beta B\mathbf{x} \quad \text{and} \quad (A - \alpha B)\bar{\mathbf{x}} = -\imath\beta B\bar{\mathbf{x}}.$$

Based on these equations, we get that $(\alpha, \mathbf{x}\otimes\bar{\mathbf{x}})$ is an eigenpair of (5):

$$\begin{aligned}[(A - \alpha B)\otimes B + B\otimes(A - \alpha B)]\,(\mathbf{x}\otimes\bar{\mathbf{x}}) &= (A - \alpha B)\mathbf{x}\otimes B\bar{\mathbf{x}} + B\mathbf{x}\otimes(A - \alpha B)\bar{\mathbf{x}} \\ &= (\imath\beta B\mathbf{x}\otimes B\bar{\mathbf{x}}) + (B\mathbf{x}\otimes(-\imath\beta)B\bar{\mathbf{x}}) = 0.\end{aligned}$$

Similarly, we can prove that $(\alpha, \bar{\mathbf{x}}\otimes\mathbf{x})$ is also an eigenpair of (5). For a real λ, that is, $\lambda = \alpha$, we use $\beta = 0$ giving us $(A - \alpha B)\mathbf{x} = 0$ from which we can again deduce that $(\lambda, \mathbf{x}\otimes\mathbf{x})$ is an eigenpair of (5).

To prove the other direction we will simplify the problem, by multiplying (5) with $B^{-1}\otimes B^{-1}$. We get equivalence of (5) with

$$\begin{aligned}\frac{1}{2}(B^{-1}A\otimes I + I\otimes B^{-1}A)\mathbf{z} &= \gamma\mathbf{z}, \\ \frac{1}{2}(R\otimes I + I\otimes R)(Q^H\otimes Q^H)\mathbf{z} &= \gamma(Q^H\otimes Q^H)\mathbf{z},\end{aligned}$$

where $B^{-1}A = QRQ^H$ is the Schur decomposition, Q unitary, R upper triangular. The eigenvalues of $(R\otimes I + I\otimes R)$ equal all possible combinations $\lambda_i + \lambda_j$ with λ_i, λ_j eigenvalues of (4). Moreover, also the structure of the eigenvectors is a consequence of this factorization. ∎

Of course, (5) can have complex eigenvalues. Any linear combination of eigenvalues of (4), λ_1 and λ_2, gives rise to a complex $\gamma = \frac{1}{2}(\lambda_1 + \lambda_2)$. These complex eigenvalues can clearly have an impact on the convergence behavior of the method. Consider the situation where (4) only has stable eigenvalues λ, i.e. all λ have negative real parts. In this case, the eigenvalue couple $\lambda, \bar{\lambda}$ nearest the imaginary axis produces a real γ. This is the eigenvalue of (5) closest to zero.

When (4) also has unstable eigenvalues, which means lying in the right-half plane, it is possible that the eigenvalue γ of (5) nearest zero is associated with two eigenvalues of (4) on both sides of the imaginary axis. Nevertheless, one can fairly easily identify these 'false' results by looking at the structure of the eigenvectors, for the stable case they are generated by $\mathbf{z} = \mathbf{x} \otimes \bar{\mathbf{x}}$ and $\bar{\mathbf{z}}$, and for the unstable case by $\mathbf{x} \otimes \mathbf{y}$ and $\mathbf{y} \otimes \mathbf{x}$. Moreover, the 'false' eigenvalues are usually complex.

We note that in the case of stability analyses of steady states, the number of eigenvalues on the right of the imaginary axis usually is small. Nonetheless, we introduce a subspace iteration method in order to compute more than one eigenvalue of (5) to avoid failing to compute the correct eigenvalue.

It is also possible that γ has multiplicity larger than two, even though the eigenvalues of (4) are all simple. For example, when there is one pair $\alpha \pm \imath\beta$ and one real eigenvalue α. The algorithm presented here does not distinguish these cases. In [14] a solution to this difficulty is proposed by only computing skew-symmetric eigenvectors, i.e. of the form $x \otimes y - y \otimes x$ instead of the symmetric vector $x \otimes y + y \otimes x$. A similar structure preserving algorithm can be developed here, eliminating thereby the eigenvectors coming from purely real λ's.

2.2. Lyapunov eigenvalue problem

The Kronecker eigenvalue problem (5) is closely related to a so-called Lyapunov eigenvalue problem. Consider Z an $n \times n$ matrix, the $\mathrm{vec}\,(\cdot)$ operator stacks all columns of the matrix Z under each other. We get the following equivalent eigenvalue problems with $\mathrm{vec}\,(Z) = \mathbf{z}$:

$$\begin{aligned} (A \otimes B + B \otimes A)\,\mathrm{vec}\,(Z) &= 2\gamma(B \otimes B)\,\mathrm{vec}\,(Z)\,, \\ BZA^T + AZB^T &= 2\gamma BZB^T\,. \end{aligned} \tag{6}$$

We will refer to the second problem as the Lyapunov eigenvalue problem, where we are interested in the γ's and matrices Z satisfying Equation (6). We will call the matrices Z 'eigenmatrices'.

Strictly speaking, we should call (6) a Sylvester eigenvalue problem, as Z can be nonsymmetric, but since we can restrict to symmetric solutions, we call this the Lyapunov eigenvalue problem.

It is interesting to remark that the corresponding eigenmatrices Z have low rank. Based on Theorem 1 we get that the eigenvectors $\mathbf{z}$ of the Kronecker eigenvalue problem are of the form $\mathbf{z} = \xi_1\mathbf{x} \otimes \mathbf{y} + \xi_2\mathbf{y} \otimes \mathbf{x}$, with ξ_1, ξ_2 two parameters. This gives us $\mathrm{vec}\,(Z) = \mathbf{z}$ with $Z = \xi_1\mathbf{y}\mathbf{x}^T + \xi_2\mathbf{x}\mathbf{y}^T$, which is of rank 2. In case the eigenvalue γ corresponds to a real λ, the associated eigenmatrix is symmetric, namely $\mathbf{x}\mathbf{x}^T$. In the other case, the eigenvalues γ are double and have associated non parallel eigenvectors $\mathbf{x}$ and $\mathbf{y}$. The eigenvectors span therefore an invariant subspace of dimension 2. Considering the Lyapunov eigenvalue problem, one can construct a symmetric eigenmatrix and a skew-symmetric eigenmatrix generating the dimension 2 subspace of eigenmatrices. Both the eigenmatrices are of rank 2, the symmetric one equals $\mathbf{y}\mathbf{x}^T + \mathbf{x}\mathbf{y}^T$ and the skew-symmetric one equals $\mathbf{y}\mathbf{x}^T - \mathbf{x}\mathbf{y}^T$.

In section 3 we present the numerical methods for solving the Kronecker eigenvalue problem and the associated Lyapunov eigenvalue problem. We first review the inexact inverse iteration method from [9] in §3.1 (see Algorithm 1). Inverse iteration applied to the Lyapunov eigenvalue problem requires the solution of a Lyapunov equation each iteration step. Since this equation is solved iteratively, we obtain an inexact inverse iteration. A detailed description of the method can be found in [9]. To reduce the computational cost an additional projection is added to the inexact inverse iteration. This is presented in §3.2 (see Algorithm 2). As an extension, we present inverse subspace iteration in §3.3, allowing one to iterate on several vectors simultaneously. This method is conceptually similar, but becomes slightly more complicated in the Lyapunov setting (see Algorithms 3 and 4). In the development of these algorithms, we mix the Kronecker and Lyapunov forms depending on which form is most convenient for the theory. In Section 4, it is shown that Algorithm 4 produces Ritz pairs identical to the ones from the implicitly restarted Arnoldi method with properly selected shifts.

3. Inverse subspace iteration with projection

Since we are interested in the eigenvalues of (4), with real part α closest to zero, we can apply inverse iteration on the Kronecker or Lyapunov eigenvalue problem.

Given a random starting vector $\mathbf{y}_0 \in \mathbb{R}^{n^2}$, inverse iteration computes iteratively normalized vectors $\mathbf{y}_j = \tilde{\mathbf{y}}_j / \|\tilde{\mathbf{y}}_j\|_2$, where

$$\frac{1}{2}(A \otimes B + B \otimes A)\tilde{\mathbf{y}}_j = (B \otimes B)\mathbf{y}_{j-1}, \quad \text{for } j \geq 1. \tag{7}$$

Under mild conditions, which are normally satisfied by random starting vectors (see e.g., [13]), the vector $\mathbf{y}_i$ converges to the eigenvector belonging to the eigenvalue of (5) closest to zero. We now rewrite the Kronecker equation to make the link with implicitly restarted Arnoldi easier to derive. Recall that we assume that A is invertible. By multiplying (7) on the left by $A^{-1} \otimes A^{-1}$, we obtain

$$\frac{1}{2}(I \otimes S + S \otimes I)\tilde{\mathbf{y}}_j = (S \otimes S)\mathbf{y}_{j-1} \quad \text{for } j \geq 1.$$

where $S = A^{-1}B$. In the remainder of the text, we will frequently refer to the matrix S since it will simplify establishing the link with the implicitly restarted Arnoldi method.

3.1. Inverse iteration

We now discuss the inverse iteration method for (6). Practically, it is, computationally inconvenient to work with $n^2 \times n^2$ matrices and vectors of length n^2 since A and B are already assumed to be large. Moreover, we also know that the desired eigenvector needs to be a sum of at most two tensor-decomposable vectors ($2n$ parameters for each tensor decomposable vector) and hence only needs $4n$ parameters instead of n^2.

Translating the above inverse iteration procedure to the Lyapunov eigenvalue problem, we get

$$\frac{1}{2}(B\tilde{Y}_j A^T + A\tilde{Y}_j B^T) = (BY_{j-1}B^T) \quad \text{for } j \geq 1.$$

Using S this becomes

$$\frac{1}{2}(S\tilde{Y}_j + \tilde{Y}_j S^T) = (SY_{j-1}S^T) \quad \text{for } j \geq 1.$$

Take a starting matrix $Y_0 \in \mathbb{R}^{n \times n}$ (Y_j is a normalized version of $\tilde{Y}_j$). By Theorem 1, we know that, for each (real) eigenvalue, there exists a symmetric eigenmatrix, hence we will assume the solution to be symmetric and only search for symmetric solutions of (6). Solving the Lyapunov equation is as expensive as solving the corresponding system in the Kronecker setting. Especially the storage of n^2 parameters for the matrix Y_i is too expensive since we know that the resulting eigenmatrix Y_i has rank at most two. To reduce the computational complexity, we will approximate each Y_i by a low rank matrix of specified rank r. As Y_i is real symmetric, the best approximation for a given rank r is obtained by approximating Y_i using a partial eigendecomposition, consisting of the r dominant eigenvalues. The matrix Y_j is thus not stored as a dense $n \times n$ matrix but in factored form $Y_j = W_j D_j W_j^T$, where W_j has orthonormal columns and D_j is a diagonal matrix. The solution techniques we will use in this article, generate solutions in low-rank factored form: $\tilde{Y}_j = W_j \tilde{D}_j W_j^T$. As normalization, we use $D_j = \tilde{D}_j / \|\tilde{D}_j\|_F$. Generically r is taken larger than 2, not to endanger or to slow down too much the convergence. Note that starting even with a rank one right-hand side Y_0, may lead to a high rank Y_1 after truncation of the smallest eigenvalues of Y_1. After a few iterations, when the iterates start converging to an eigenmatrix, a good approximation by a low rank matrix is possible without much loss of precision, as the eigenmatrices have rank at most two.

The problem, as it is considered here, has large n. There exists a variety of iterative solvers for the Lyapunov problem which can be found for example in [15–22] for Krylov based methods, [23, 24] for ADI type methods, and [25] for the Smith method. Overviews are presented in [26, 27]. A key property of many of these algorithms is that they control the rank of the solution by a built-in truncation step.

We will use the block Arnoldi method for reducing the dimensions of the Lyapunov equation, because of the link with the implicitly restarted Arnoldi method. The block size is equal to the rank of Y_{j-1}. For obvious reasons, this limits the rank of the solution to the number of Krylov vectors. In order to limit the cost of successively solving

Lyapunov equations with a right-hand side of large rank, we can even reduce further the rank of the solution by truncating the small eigenvalues of $\tilde{Y}_j$ as we discussed before. This is not always possible, in general, but for many operators, low rank solutions can be expected [28, 29].

The following algorithm implements the inverse iteration method as depicted above. The solutions of the Lyapunov equation are denoted by $\tilde{Y}_j$, and their normalizations by Y_j.

Algorithm 1 (Inverse Iteration on the Lyapunov Equation).

1. Given $Y_0 = W_0 D_0 W_0^T$, W_0 a column vector with $\|W_0\|_2 = 1$ and $\|D_0\|_F = 1$. Set $j = 0$.
2. While not converged
   (a) Solve $\tilde{Y}_{j+1}$ in factored form $\tilde{Y}_{j+1} = W_{j+1} D_{j+1} W_{j+1}^T$ from

$$\frac{1}{2}(S\,\tilde{Y}_{j+1} + \tilde{Y}_{j+1} S^T) = S\,Y_j S^T.$$

   (b) Normalize $\tilde{Y}_{j+1}$ and store it in Y_{j+1}.
   (c) Increase j: $j = j + 1$.

To check whether the method has converged we first compute the corresponding approximate eigenvalue γ as

$$\gamma = -\frac{\operatorname{trace}(D_j^2 \tilde{S}_j^T + D_j \tilde{S}_j D_j)}{\operatorname{trace}(2 D_j \tilde{S}_j D_j \tilde{S}_j)}, \tag{8}$$

where $\tilde{S}_j = W_j^T S\, W_j$. Equation (8) is based on the Rayleigh quotient for Kronecker products; see e.g., [30]. To check for convergence, we compute the associated α_j, β_j and $\mathbf{x}_j$ from the projected small system $W_j^T(I - \alpha_j S) W_j \mathbf{x}_j = \imath\beta_j W_j^T S\, W_j \mathbf{x}_j$ and check the residual norm

$$\|(I - \alpha_j S) W_j \mathbf{x}_j - \imath\beta_j S\, W_j \mathbf{x}_j\|_2.$$

Computing this residual norm is not a bottle-neck since it is much faster than solving the corresponding Lyapunov equation.

When running this algorithm, the rank of the successive iterates Y_j can be large. Restricting the rank to a certain threshold r is surely helpful in order to reduce the computational cost of the Lyapunov solver. From experiments, we found that r can be ten or larger, which makes it potentially impractical for real life applications and thus a rank restriction is required.

3.2. Inverse iteration with projection

In [9], the Lyapunov eigenvalue problem (6) is projected on the Krylov space used by the Lyapunov solver. Let V_{j+1} denote the basis vectors obtained from the block Arnoldi method for computing $\tilde{Y}_{j+1}$ in Algorithm 1. Define

$$\tilde{S}_{j+1} = V_{j+1}^T S\, V_{j+1} \in \mathbb{R}^{k \times k}. \tag{9}$$

Then we solve the order k Lyapunov eigenvalue problem

$$\frac{1}{2}(\tilde{Z}_{j+1} \tilde{S}_{j+1}^T + \tilde{S}_{j+1} \tilde{Z}_{j+1}) = \tilde{\gamma}_{j+1} \tilde{S}_{j+1} \tilde{Z}_{j+1}^T \tilde{S}_{j+1}^T. \tag{10}$$

The eigenmatrices have at most rank two. The corresponding Ritz eigenmatrices for the large scale Lyapunov eigenvalue problem (6) also have rank two: $Z_{j+1} = V_{j+1} \tilde{Z}_{j+1} V_{j+1}^T$. The advantage of the projection step is twofold: faster convergence is expected than with inverse iteration and the right-hand sides have rank two (at most) [9]. The projected equation (10) can be solved by the QZ-method, which has a complexity of the order k^6, or if this would be too expensive, inverse iteration using the Bartels and Stewart [31] direct linear system solver.

Algorithm 2 (Inverse Iteration with Projection on the Lyapunov Equation).

1. Take $Z_0 = W_0 D_0 W_0^T$. Set $j = 0$.

2. While not converged
   (a) Compute the k Krylov vectors, denoted by V_{j+1}, generated for solving

$$\frac{1}{2}(S\tilde{Y}_{j+1} + \tilde{Y}_{j+1}S^T) = SZ_jS^T. \tag{11}$$

   (b) Solve the projected Lyapunov eigenvalue problem (10) and let $\tilde{Z}_{j+1}$ be the eigenmatrix associated with the eigenvalue nearest zero. In practice, we decompose $\tilde{Z}_{j+1} = \tilde{W}_{j+1}D_{j+1}\tilde{W}_{j+1}^T$ with $\tilde{W}_{j+1} \in \mathbb{R}^{k\times r}$ with $r = 1$ or $r = 2$.
   (c) Compute the Ritz eigenmatrix in factored form $Z_{j+1} = W_{j+1}D_{j+1}W_{j+1}^T$ with $W_{j+1} = V_{j+1}\tilde{W}_{j+1}$.
   (d) Increase j: $j = j + 1$

The stopping criterion of this algorithm is the same as for the previous algorithm without the projection step. Note that the block Krylov method for solving (11) does not require D_j. This fact will be used for the extension to subspace iteration.

3.3. Inverse subspace iteration

As discussed before, inverse iteration only enables convergence towards a single eigenvalue. Assume now for robustness that we want to compute several eigenvalues simultaneously. To achieve this, subspace iteration is used. For simplicity of notation, we will now change to the Kronecker formulation. Instead of iterating on a single vector $\mathbf{y}_j$, we will now iterate on several vectors at the same time. Given ℓ starting vectors $\left[\mathbf{y}_0^{(1)}, \ldots, \mathbf{y}_0^{(\ell)}\right]$, where the superscript (i) denotes the ith vector, subspace iteration is of the following form. We solve for $\left[\tilde{\mathbf{y}}_{j+1}^{(1)}, \ldots, \tilde{\mathbf{y}}_{j+1}^{(\ell)}\right]$,

$$\frac{1}{2}(I \otimes S + S \otimes I)\left[\tilde{\mathbf{y}}_{j+1}^{(1)}, \ldots, \tilde{\mathbf{y}}_{j+1}^{(\ell)}\right] = (S \otimes S)\left[\mathbf{y}_j^{(1)}, \ldots, \mathbf{y}_j^{(\ell)}\right] \quad \text{for } j \geq 0. \tag{12}$$

The columns of $\left[\mathbf{y}_{j+1}^{(1)}, \ldots, \mathbf{y}_{j+1}^{(\ell)}\right]$ are orthonormalized by using, for instance, Gram-Schmidt orthogonalization. Let us denote the matrices containing these vectors as bold capital letters: $\boldsymbol{Y}_{j+1}$ and $\tilde{\boldsymbol{Y}}_{j+1}$.

Algorithm 3 (Subspace iteration on the Kronecker problem).

1. Let $\boldsymbol{Y}_0 \in \mathbb{R}^{n^2\times\ell}$ denote a starting matrix with orthonormal columns. Set $j = 0$.
2. While not converged
   (a) Solve $\tilde{\boldsymbol{Y}}_{j+1}$ from $1/2(I \otimes S + S \otimes I)\tilde{\boldsymbol{Y}}_{j+1} = (S \otimes S)\boldsymbol{Y}_j$.
   (b) Orthonormalize the columns of $\tilde{\boldsymbol{Y}}_{j+1}$ to get $\boldsymbol{Y}_{j+1}$.
   (c) Increase j: $j = j + 1$.

The structure of the eigenvectors is not exploited here and also the fact that dimensions are squared, is computationally very inconvenient. Moreover, the algorithm does not respect the fact the columns of $\boldsymbol{Y}$ transform to a symmetric $n \times n$ matrix. To overcome this problem, we switch back to the Lyapunov setting. The algorithm is similar to the previous one. The major difference is that ℓ Lyapunov equations need to be solved, one for each Ritz vector, and that the resulting Krylov spaces are added together in a new subspace, which is then used for the projection. In the projection phase, ℓ Ritz pairs are computed.

Let the ℓ Ritz eigenmatrices be $W_j^{(i)}D_j^{(i)}W_j^{(i)^T}$ for $i = 1, \ldots, \ell$. Instead of solving a Lyapunov equation for each eigenmatrix, we can solve one Lyapunov equation with right-hand side matrix $W_jD_jW_j^T$ where the columns of W_j span all columns of $W_j^{(1)}, \ldots, W_j^{(\ell)}$. The Krylov space generated by the block Arnoldi method on W_j is the sum of the Krylov spaces started with $W_j^{(i)}$, $i = 1, \ldots, \ell$. This produces one large block Krylov space for all right-hand sides together. Note that the matrix D_j is not used by the block Arnoldi method and is unimportant here. At first sight, there is no benefit to solving all Lyapunov equations together. However, dependencies in the blocks may occur so that the Krylov blocksize may be reduced during the execution of the block Arnoldi method [16]. The ℓ eigenmatrices are computed from an order k Lyapunov eigenvalue problem (see Equation (10)). The starting vectors for the next iteration are extracted from those eigenmatrices.

The following algorithm presents this idea.

Algorithm 4 (Subspace iteration on the Lyapunov problem).

1. Take $Z_0 = W_0 D_0 W_0^T$, with $W_0 \in \mathbb{R}^{n\times 1}$. Set $j = 0$.
2. While not converged
   (a) Compute V_{j+1} of dimension $n\times k$ from an iterative method (block Arnoldi for example) with starting block of vectors W_j. This is related to solving a Lyapunov equation with right-hand side $Z_j = W_j D_j W_j^T$.
   (b) Compute ℓ eigenpairs $(\tilde{\gamma}_i, \tilde{Z}^{(i)})$ for $i = 1, \ldots, \ell$ of the Lyapunov eigenvalue problem (10).
   (c) Compute the associated Ritz eigenmatrices in factored form $Z_j^{(i)} = W_{j+1}^{(i)} \tilde{D}^{(i)} (W_{j+1}^{(i)})^T$ for $i = 1, \ldots, \ell$.
   (d) Compute $W_{j+1} \in \mathbb{R}^{n\times\ell}$ so that all columns of $W_{j+1}^{(i)}$, $j = 1, \ldots, \ell$ are spanned by the columns of W_{j+1}.

Note that the matrix D_j is not used in this algorithm. Also note that in the first iteration, we start with the rank one matrix Z_0. After the first iteration, we keep ℓ Ritz pairs. Starting with a rank one matrix is required to make the connection with IRAM later on.

4. The relation with implicitly restarted Arnoldi

In this section, we will briefly recapitulate the (implicitly restarted) Arnoldi method [5, 7] and then consider the link with the Lyapunov eigenvalue problem.

The Arnoldi procedure is well-known for generating a sequence of orthonormal vectors, such that the resulting projected matrix is of Hessenberg form [13, 32, 33]. Let us briefly recapitulate the construction of the orthonormal vectors. Let S be an $n \times n$ matrix, $\mathbf{v}_1$ a starting vector. The Krylov space of dimension k, with starting vector $\mathbf{v}_1$ is defined as $\mathcal{K}_k(S, \mathbf{v}_1) = \text{span}\{\mathbf{v}_1, S\mathbf{v}_1, S^2\mathbf{v}_1, \ldots, S^{k-1}\mathbf{v}_1\}$. The Arnoldi procedure iteratively generates an orthonormal basis for the Krylov subspaces growing at each iteration step. In each iteration, $S\mathbf{v}_j$ is orthogonalized against the previously computed orthonormal vectors and stored in $\mathbf{v}_{j+1}$, expressed as follows, with $h_{i,j}$ being the Gram-Schmidt coefficients:

$$S\mathbf{v}_j - h_{1,j}\mathbf{v}_1 + \cdots + h_{j,j}\mathbf{v}_j = h_{j+1,j}\mathbf{v}_{j+1},$$

which, for $j = 1, \ldots, k$ (let $k < n$), can be rewritten in matrix language as

$$S V_k = V_k H_k + h_{k+1,k}\mathbf{v}_{k+1}\mathbf{e}_k^T, \tag{13}$$

where $V_k = [\mathbf{v}_1, \ldots, \mathbf{v}_k]$ and $H_k = [h_{i,j}]$ is a $k \times k$ proper upper Hessenberg matrix. Equation (13) is called the recurrence relation, since it gives the relation between successive iteration vectors. It is also called an order k Arnoldi factorization.

Under some mild conditions, the eigenvalues of the Hessenberg matrix H_k (named Ritz-values) approximate the well-separated eigenvalues of the matrix S [8, 34]. Often, the dominant eigenvalues (i.e. the ones of largest magnitude) are well-separated eigenvalues.

Let $H_k\mathbf{z} = \theta\mathbf{z}$. Then θ is called a Ritz value and $\mathbf{y} = V_k\mathbf{z}$ an associated Ritz vector. The residual $\mathbf{r} = S\mathbf{y} - \theta\mathbf{y}$ can be computed cheaply from (13) as $\mathbf{r} = h_{k+1,k}\mathbf{v}_{k+1}\mathbf{e}_k^T\mathbf{z}$ and the residual norm $\|\mathbf{r}\| = h_{k+1,k}|\mathbf{e}_k^T\mathbf{z}|$. The stopping criterion of the Arnoldi method (and IRAM) is usually based on the residual norm, i.e. the method is stopped when $\|\mathbf{r}\|$ is below a prescribed tolerance.

Unfortunately, it may happen that a large number of iterations is required, before an accurate solution is obtained. Storing a large number of iteration vectors becomes prohibitive. This was the motivation for the implicitly restarted Arnoldi method, which is explained in the following section.

4.1. Implicitly restarted Arnoldi

In the implicitly restarted Arnoldi procedure, we do not start from scratch with a new starting vector, but we shrink the existing Krylov subspace to a smaller dimension by removing unwanted directions from the subspace. Globally, the implicitly restarted Arnoldi procedure shrinks and expands a Krylov subspace on every restart.

Reducing the dimension of the Krylov subspace from k to p is done by performing $k - p$ steps of the shifted QR-method on the Hessenberg matrix H_k. The orthogonal transformation is applied to the Krylov vectors. The method starts by performing $k - p$ shifted QR steps on H_k with shifts $\nu_1, \ldots, \nu_{k-p}$. Let Q denote the orthogonal transformation

that accumulates the orthogonal transformations of the QR steps on H_k. Then define $V_k^\star = V_k Q$ and $H_k^\star = Q^* H_k Q$. This results in the modified recurrence relation

$$S V_k^\star - V_k^\star H_k^\star = h_{k+1,k} v_{k+1} e_k^T Q$$

Truncating the last $k - p$ columns of the modified recurrence relation leads to the order p Arnoldi factorization

$$S V_p^+ - V_p^+ H_p^+ = h_{p+1,p}^+ v_{p+1}^+ e_p^T$$

with V_p^+ the first p columns of $V_k^\star$, H_p^+ the leading $p \times p$ submatrix of $H_k^\star$, and v_{p+1}^+ and $h_{p+1,p}^+$ so that $\|v_{p+1}^+\|_2 = 1$ and $h_{p+1,p}^+ v_{p+1}^+ = v_{p+1}^\star h_{p+1,p}^\star + h_{k+1,k} v_{k+1} q_{k,p}$. See [5] for the technical details. The last $k - p$ Krylov vectors are thus truncated. An important property of this QR-based reduction procedure is that the remaining vectors still span a Krylov subspace, but one of smaller dimension. By a proper selection of shifts, undesired directions are removed from the Krylov space and promising directions are enhanced. Roughly speaking, we can state that the shifts should be picked close to the eigenvalues we do not want to keep. Unfortunately, it is not always clear which shifts should be chosen for the shifted QR-method, nor what p should be taken.

The implicitly restarted Arnoldi method applied on a matrix S is of the following form.

Algorithm 5 (Implicitly restarted Arnoldi method (IRAM)).

1. Let $\mathbf{v}_1$ denote a starting vector.
2. Build the order k Arnoldi factorization, starting from $\mathbf{v}_1$.
3. While not converged to the desired eigenvalues
   (a) Select $k - p$ shifts $\nu_1, \ldots, \nu_p$.
   (b) Apply a QR-step for each of the shifts ν_i ($1 \le i \le p$) on the Hessenberg matrix H_k, apply the accumulated orthogonal transformations to the Krylov vectors and remove the $k - p$ trailing vectors from the Krylov space.
   (c) Expand the existing Krylov space of order p to a space of dimension k by $k - p$ Arnoldi steps.

Convergence of this method is tested by computing the Ritz-values and checking if their residual norms are smaller than a prescribed tolerance.

The details of the method and its derivation can be found in Sorensen's work [5, 7]. Many choices of shifts are possible. We will use a selection of Ritz-values as shifts, so-called exact shifts. Assume matrix H_k has Ritz-values $\theta_1, \ldots, \theta_k$ and associated Ritz-vectors $\mathbf{y}_1, \ldots, \mathbf{y}_k$. Assume we want to keep $\theta_1, \ldots, \theta_p$ and directions $\mathbf{y}_1, \ldots, \mathbf{y}_p$. Applying $k - p$ QR-steps in the implicit method with the remaining Ritz-values $\theta_{p+1}, \ldots, \theta_k$ as shifts, filters out these Ritz vectors and leaves us with the Krylov subspace $\text{span}\{\mathbf{y}_1, \ldots, \mathbf{y}_p\}$. Expanding now again this Krylov subspace by the Arnoldi procedure gives us a new subspace spanned by k vectors. Moreover, it is proved in [6] that all the following subspaces

$$\text{span}\{\mathbf{y}_1, \ldots, \mathbf{y}_p, S\mathbf{y}_i, S^2\mathbf{y}_i, \ldots, S^{k-p-1}\mathbf{y}_i\} \quad \text{for } 1 \le i \le p, \tag{14}$$

span the same Krylov subspace of dimension k. It was observed in [5, 6] that restarting Arnoldi does not necessarily increase the number of iterations compared to a full Arnoldi process when exact shifts are used. The reason is that the shifts that are close to eigenvalues 'deflate' those eigenvalues from the subspace.

Hence the impact of the implicitly restarted Arnoldi method is two-fold: firstly, it filters out all undesired directions and, secondly, it enhances the other directions by subspace iteration. Indeed, from (14), we see that the p power sequences $\mathbf{y}_i, S\mathbf{y}_i, S^2\mathbf{y}_i, \ldots, S^{k-p-1}\mathbf{y}_i$, $i = 1, \ldots, p$ lie in the Krylov space.

Another choice of shift is a zero shift. In that case, the subspace dimension is also reduced by one, i.e. the order k Arnoldi factorization

$$S V_k - V_k H_k = h_{k+1,k} \mathbf{v}_{k+1} \mathbf{e}_k^T$$

is transformed to the order $k - 1$ Arnoldi factorization

$$S V_{k-1}^+ - V_{k-1}^+ H_{k-1}^+ = h_{k,k-1}^+ \mathbf{v}_k^+ \mathbf{e}_{k-1}^T$$

where $\text{Range}(V_k^+) = \text{Range}(S V_k)$ with $V_k^+ = [V_{k-1}^+, \mathbf{v}_k^+]$. We can combine both exact shifts and a zero shift.

Theorem 2. *Given the order k Arnoldi factorization* (13)*, let* $(\theta_j, \mathbf{y}_j)$ *for* $j = 1, \ldots, k$ *be the k Ritz pairs. Then applying* $k-p$ *exact shifts and one zero shift leads to an Arnoldi factorization of order* $p-1$*. By performing* $k-p+1$ *additional Arnoldi steps, we obtain the Arnoldi factorization* (13) *where the columns of* V_k *span*

$$\{S\mathbf{y}_1, \ldots, S\mathbf{y}_p, S^2\mathbf{y}_i, \ldots, S^{k-p}\mathbf{y}_i\} \quad \textit{for any } i : 1 \le i \le p. \tag{15}$$

Proof. From [5, 6], applying the exact shifts $\theta_{p+1}, \ldots, \theta_k$ produces the order p Arnoldi factorization

$$S V_p^+ - V_p^+ H_p^+ = h_{p+1,p}^+ \mathbf{v}_{p+1}^+ \mathbf{e}_p^T$$

where $\text{Range}(V_p^+) = \text{Range}(\mathbf{y}_1, \ldots, \mathbf{y}_p)$. With one additional shift at zero, we obtain the Arnoldi factorization

$$S W_{p-1} - W_{p-1} G_{p-1} = g_{p,p-1} \mathbf{w}_p \mathbf{e}_{p-1}^T,$$

where $\text{Range}([W_{p-1}, \mathbf{w}_p]) = \text{Range}(S V_p^+)$ [10]. Performing $k - p + 1$ additional Arnoldi steps produces an order k Arnoldi factorization (13). Since all powers $S^j \mathbf{y}_i$, $i = 1, \ldots, p$, $j = 1, \ldots, k-p$ lie in the Krylov space, the columns of V_k span (15). ∎

For the solution of generalized eigenvalue problems (3), the Arnoldi method is often applied to $S = A^{-1}B$ which favors the convergence of the eigenvalues near zero. Therefore it is assumed that all eigenvalues near the imaginary axis are sufficiently close to zero, which is usually the case. The eigenvalues θ of H_k are now approximations to eigenvalues of S. In order to find an eigenvalue of (3), we must compute λ as $\lambda = \theta^{-1}$. The connection with this method and Algorithm 4 will be given in the next subsection.

Alternatively, a shift can be used as in shift-and-invert Arnoldi, i.e. the Arnoldi method applied to $S = (A - \sigma B)^{-1} B$, which is helpful in some situations [4]. We will give an example in §5.

4.2. The solution of Lyapunov equations

Assume we have the following Lyapunov equation to solve:

$$Y S^T + S Y = S\mathbf{w}(S\mathbf{w})^T. \tag{16}$$

This problem can be solved by Arnoldi's method [15] applied to S with starting vector $S\mathbf{w}$. Suppose that V_k is the matrix of corresponding Arnoldi vectors and H_k is the Hessenberg matrix. Now consider the order k Lyapunov equation

$$\tilde{Y} H_k^T + H_k \tilde{Y} = \mathbf{e}_1 \mathbf{e}_1^T \|S\mathbf{w}\|^2$$

which is small if k is small and can be solved by the method of Bartels and Stewart [31]. We use $Y = V_k \tilde{Y} V_k^T$ as an approximate solution for (16).

When the right-hand side of (16) has rank larger than one, a block Krylov subspace method can be used [16]. This is, however, not needed as we explain in the next subsection. As we shall see, we have a right hand side that is the basis of a Krylov space. The following lemma then becomes useful.

Lemma 3. *If the columns of* $V_0 \in \mathbb{R}^{n \times \ell}$ *are a Krylov basis for* S*, then the block Arnoldi method applied to* S *with starting vectors* V_0 *produces a Krylov space where the starting vector is a linear combination of the columns of* V_0*. In a certain sense we go from a block Arnoldi to a regular Arnoldi method.*

Proof. This is a well-known property, and can be shown as follows. Since the columns of V_0 form a Krylov basis, we have that

$$S V_0 - V_0 H = R$$

where R is a rank one matrix. In the first block Arnoldi step, we orthogonalize $S V_0$ against V_0. The remaining vectors form a rank one matrix, which proves the lemma. ∎

4.3. Inverse iteration on the Lyapunov equation and IRAM

We first consider the case of $\ell = 1$, i.e., inverse subspace iteration becomes standard inverse iteration. Since we work with real matrices, we assume that we are looking for a simple and real γ. We will show that Lyapunov inverse iteration with projection (Algorithm 4) produces the same subspaces as an explicitly restarted Arnoldi method. Then, we show that, for the case $\ell > 1$, there is a connection with implicitly restarted Arnoldi with a proper selection of the shifts.

We first consider $\ell = 1$, which means inverse iteration.

Theorem 4. *Let the implicitly restarted Arnoldi method be started with the vector SW_0 from Algorithm 4 and for the restarts k shifts are taken according to Theorem 2. Assume the k in Algorithm 4 (for $\ell = 1$) and Algorithm 5 are identical and let the Lyapunov equation be solved by a Krylov method, whose subspace is generated by S. Then the Ritz vectors produced by both methods are the same for each iteration $j \geq 0$.*

Proof. In Algorithm 4, S is projected on the Krylov space. This resulting matrix is the Hessenberg matrix H_k. We will show that the Krylov spaces are the same for both Algorithms 4 and 5. Hence, the Ritz pairs are the same for both algorithms. The solution of the order k Lyapunov eigenvalue problem (10) therefore is the pair $(\theta_1, \mathbf{y}_1\mathbf{y}_1^T)$, where $\theta_1, \mathbf{y}_1$ is a Ritz pair of Algorithm 5. So, we must prove that the same Krylov spaces are built.

We prove this by induction. In the first iteration, the Krylov space for the Lyapunov solver starts with SW_0, which is also the case for Algorithm 5.

Suppose now, by induction, that the computed Ritz vector W_j (Algorithm 4) and $\mathbf{y}_1$ (Algorithm 5) are parallel for both algorithms after iteration j. We now prove that the Krylov spaces are the same at iteration $j + 1$. The order k Arnoldi factorization produces an upper Hessenberg matrix H_k. When we use the shifts, mentioned in Theorem 2, we keep a single vector, which, according to the theorem is the vector $S\mathbf{y}_1$, where $\mathbf{y}_1$ is the Ritz vector associated with θ_1. So, the Arnoldi method is explicitly restarted with $S\mathbf{y}_1$. In iteration $j + 1$ of Algorithm 4, a Lyapunov equation with right-hand side $S\mathbf{y}_1\mathbf{y}_1^T S^T$ needs to be solved. When we use, in iteration $j + 1$, the Arnoldi Lyapunov solver, explained earlier, we obtain the same order k Arnoldi factorization, since the starting vectors are identical.

This proves the theorem. ■

We now show the proof for $\ell > 1$; this means inverse subspace iteration.

Theorem 5. *Let the implicitly restarted Arnoldi method be started with the vector SW_0 from Algorithm 4 and for the restarts $k - \ell + 1$ shifts are taken according Theorem 2. Assume the k in Algorithm 4 and Algorithm 5 are identical and let the Lyapunov equation be solved by a block Krylov method, whose subspace is generated by S. Then the Ritz vectors produced by both methods are the same for each iteration $j \geq 0$.*

Proof. We prove the theorem in a similar way as Theorem 4. The first iteration is the same as for the case $\ell = 1$. So, after the first iteration, the ℓ Ritz pairs are identical for both methods.

Assume we have the same Ritz vectors $\mathbf{y}_1, \ldots, \mathbf{y}_\ell$ at the beginning of iteration $j + 1$. These vectors form a Krylov space, following the theory from §4. Following Theorem 2, the implicitly restarted Arnoldi method with zero shift produces the vectors $S\mathbf{y}_1, \ldots, S\mathbf{y}_\ell$. Since the Ritz vectors are equal for both methods, the block Krylov method for the Lyapunov equation uses starting vectors $[S\mathbf{y}_1, \ldots, S\mathbf{y}_\ell]$. As these starting vectors form a Krylov space, the block Arnoldi method reduces to the Arnoldi method following Lemma 3. This produces the same starting vectors for the Arnoldi method after the implicit restart in Algorithm 5. The remainder of the proof is almost identical to the proof of Theorem 4. ■

4.4. The case of singular B

In this section, we briefly comment on the case of singular B. This case arises in applications, e.g., the (Navier) Stokes problem. In this paper, we do not perform a full analysis, neither do we give a numerical example. We do show that in some cases, an extension of the current method to singular B is straightforward.

When B is singular, the matrices in the left- and right-hand side of (5) have a common nullspace so that all γ are eigenvalues with eigenvectors of the form $z = z_1 \otimes z_2$ where $Bz_1 = Bz_2 = 0$. The nullspace of B generates infinite eigenvalues for (4), which are physically irrelevant and whose computation should be avoided. In the literature several

shifts	k	p	restarts	λ	residual norm
(I)	10	5	10	$-0.722241 \pm 4.20999\imath$	$8.3 \cdot 10^{-5}$
(II)	10	5	10	$-0.723921 \pm 4.20894\imath$	0
(I)	20	10	5	$-0.723921 \pm 4.20894\imath$	0
(II)	20	10	5	$-0.723921 \pm 4.20894\imath$	0

Table 1: Ritz values nearest the imaginary axis for different values of k and p and choices of shifts for the Olmstead problem

techniques are proposed for the generalized eigenvalue problem, by using the B inner product and a zero implicit restart [10, 35]. These methods build Krylov spaces that lie in the range of S, which does not have components in the nullspace of $S = A^{-1}B$, which is the nullspace of B. In exact arithmetic, such Krylov spaces can be built using starting vectors that have no components in this nullspace. Such can be achieved by multiplying the starting vector by S. Alternatively, implicit restarts with a zero shift are also possible [10]. The difference with the algorithm presented here, is that in [10], the zero shift is applied right *before* the computation of the Ritz values. This reduces the rounding errors arising from the nullspace of B, as was shown in [10].

5. Numerical examples

In this section, we illustrate the implicitly restarted Arnoldi method using the classical strategy with exact shifts and the strategy with exact shifts combined with a zero shift, suggested by the solution of the Kronecker eigenvalue problem using inverse iteration. We also illustrate the solution of a parametric eigenvalue problem using subspace inverse iteration. The first example is the Olmstead model, which is a classical dynamical system, used to illustrate bifurcation analyses. The second example mimics the difficult situation of many stable eigenvalues near zero and a complex imaginary pair far away from zero, that does occur in applications.

5.1. The Olmstead model

The mathematical model represents the flow of a layer of viscoelastic fluid heated from below [36, 37]. The equations are

$$\begin{aligned} \frac{\partial u}{\partial t} &= (1-\gamma)\frac{\partial^2 v}{\partial X^2} + \gamma\frac{\partial^2 u}{\partial X^2} + \rho u - u^3 \\ \delta\frac{\partial v}{\partial t} &= u - v \end{aligned}$$

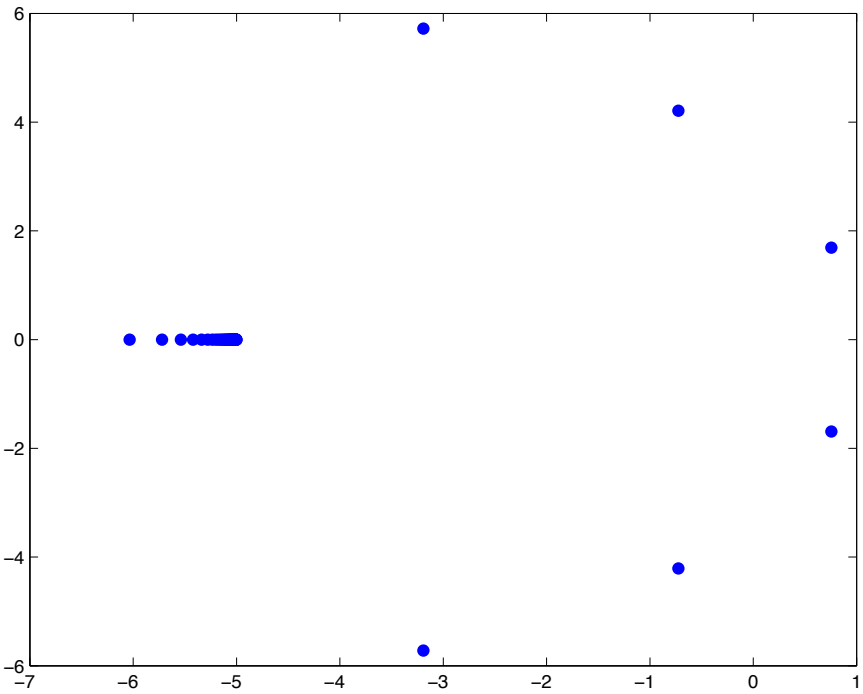
where u represents the speed of the fluid and v is related to viscoelastic forces. The boundary conditions are $u(0) = u(1) = 0$ and $v(0) = v(1) = 0$. After discretization with central differences with grid-size $h = 1/(n/2+1)$, the equations may be written as $d\mathbf{x}/dt = \mathbf{f}(\mathbf{x})$ with $\mathbf{x} = [u_1, v_1, u_2, v_2, \ldots, u_{N/2}, v_{N/2}]^T$. We consider the Jacobian $A = \partial\mathbf{f}/\partial\mathbf{x}$ for $n = 10,000$, $\delta = 2$, $\gamma = 0.1$ and $\rho = 0.6$, evaluated in the trivial steady state solution. Figure 1 shows the part of the spectrum near the imaginary axis. Most eigenvalues lie on the left of this picture far away from the imaginary axis.

As first selection (I) of shifts we used the $k - p$ eigenvalues of H_k that correspond with the left most Ritz values of (4), and as second selection (II), the $k - \ell = k - p - 1$ left most Ritz values and a zero shift. Choice (I) corresponds to the classical choice of (exact) shifts, where (II) corresponds to the method from [9]. Note that for both selections, the computational cost is of the same order for the same k and p. We see in Table 1 that with the second selection, the desired eigenvalue was found to full accuracy. The reason is that, with the first selection, the Arnoldi method first converged to the eigenvalues $0.75652 \pm 1.69189\imath$ and then started to converge to the desired eigenvalue. For the larger value of k, 20, we do not see significantly different behavior.

5.2. Purely imaginary eigenvalues

We generated an $n \times n$ matrix A with $n = 10,000$, $B = I$, such that A has eigenvalues $-1, -2, \ldots, -9998$ and the complex pair $\pm 30\imath$. That means that the eigenvalues nearest the imaginary axis are the purely imaginary pair $\pm 30\imath$. This construction simulates the physical situation in the double-diffusive convection example [38, 39].

Figure 1: Part of the spectrum of the Olmstead equation



shifts	k	p	restarts	λ	residual norm
(I)	20	10	3	$+1.37 \cdot 10^{-14} \pm 30\imath$	$2.9 \cdot 10^{-12}$
(II)	20	10	3	$-2.04 \cdot 10^{-13} \pm 30\imath$	$9.7 \cdot 10^{-13}$
(I)	10	5	10	$+7.58 \cdot 10^{-14} \pm 30\imath$	$1.3 \cdot 10^{-10}$
(II)	10	5	10	$+3.21 \cdot 10^{-13} \pm 30\imath$	$2.4 \cdot 10^{-11}$

Table 2: Ritz values nearest the imaginary axis for different values of k and p and choices of shifts for the problem with purely imaginary eigenvalues

In a first test, we compared the implicitly restarted Arnoldi method with $k = 20$ and $k = 10$ Krylov vectors and $p = 10$ and $p = 5$ vectors after the restart, respectively. As first selection (I) of shifts we used the $k - p$ eigenvalues of H_k that correspond with the left most Ritz values of (4), and as second selection (II), the $k - p - 1$ left most Ritz values and a zero shift. Note that for both selections, the computational cost is of the same order for the same k and p. Table 2 shows the Ritz values and their residual norms before each implicit restart for different values of k and p. We see that the convergence behavior is very similar for both choices of shifts. It should be noted that when a problem has eigenvalues on the imaginary axis, the Lyapunov equation solved in the inverse iteration method does not have a unique solution. This may be a concern in a practical implementation of Algorithm 4 using a Lyapunov solver.

We now perform the same computation, but using a shift on A, i.e. we shift the matrix into $A - \sigma I$ with $\sigma = 10$. We then compute the eigenvalues nearest the vertical line through 10. Table 3 shows the results for the same parameters as the previous runs. Note that the residual norms are for the shift-and-invert transformation. As σ is now 10, a smaller residual norm does not necessarily imply a more accurate eigenvalue. However, we notice that the real part of the Ritz value has one more accurate digit with $\sigma = 10$.

shifts	k	p	restarts	λ	residual norm
(I)	20	10	3	$+3.02 \cdot 10^{-14} \pm 30\imath$	$6.2 \cdot 10^{-16}$
(II)	20	10	3	$+3.55 \cdot 10^{-15} \pm 30\imath$	$7.1 \cdot 10^{-16}$
(I)	10	5	10	$-3.55 \cdot 10^{-15} \pm 30\imath$	$3.6 \cdot 10^{-18}$
(II)	10	5	10	$+2.49 \cdot 10^{-14} \pm 30\imath$	$1.2 \cdot 10^{-17}$

Table 3: Ritz values nearest the imaginary axis for different values of k and p and choices of shifts for the problem with purely imaginary eigenvalues, using $\sigma = 10$

α	residual	α	residual
0.600251	$1.8 \cdot 10^{-1}$	0.600251	$1.3 \cdot 10^{0}$
$0.2788 + 0.4861\imath$	$8.1 \cdot 10^{-3}$	$0.2788 - 0.4861\imath$	$6.3 \cdot 10^{-1}$
1.44783	$8.5 \cdot 10^{-7}$	−1.51304	$5.2 \cdot 10^{-4}$
1.44783	$1.9 \cdot 10^{-10}$	−1.51304	$9.8 \cdot 10^{-4}$

Table 4: Convergence behavior for the computation of four eigenvalues of the parameterized Olmstead problem

5.3. Inverse subspace iteration for a parameterized eigenvalue problem

Recall the Olmstead equation from Section 5.1. We consider here the parameterized Jacobian $A + \alpha B$ for $n = 20,000$, $\delta = 2$, $\gamma = 0.1$ with parameter $\alpha = \rho - 0.6$ where $\rho \in [0.6, 5]$, evaluated in the trivial steady state solution. In this example, we do not compute the eigenvalues nearest the imaginary axis, but we want to compute the values of ρ for which we have purely imaginary eigenvalues. That is, we want to compute α so that $\imath\beta$ is an eigenvalue of $(A + \alpha B)\mathbf{x} = \imath\beta\mathbf{x}$.

We used Algorithm 4 with $\ell = 4$ and Krylov subspace dimension $k = 40$. Table 4 shows the computed α's and the residual norms for four iterations of the method. Each line corresponds to an iteration (or restart) and each column to an eigenvalue. Each of the printed eigenvalues α have multiplicity two. The double eigenvalues have the same residual norms, so we do not print four columns.

6. Conclusions

In this article, an alternative approach, based on the Kronecker and Lyapunov setting was proposed for computing generalized eigenvalues closest to the imaginary axis. The method was based on inverse subspace iteration applied to the Lyapunov system. An extra projection step was required to restrict the rank of the intermediate solutions. The link with implicitly restarted Arnoldi enabled us to theoretically predict the convergence of the method, relying on established convergence theory. Both methods perform subspace iteration on a selection of Ritz vectors obtained from a projection step.

The results in this article serve as a first step towards a better understanding of the more general method as proposed by Meerbergen and Spence in [9].

[1] K. Cliffe, T. Garratt, A. Spence, Calculation of eigenvalues of the discretised Navier-Stokes and related equations, in: J. Whiteman (Ed.), The mathematics of finite elements and applications VII MAFELAP, Academic Press, 1990, pp. 479–486.

[2] T. Garratt, G. Moore, A. Spence, Two methods for the numerical detection of Hopf bifurcations, in: R. Seydel, F. Schneider, H. Troger (Eds.), Bifurcation and Chaos : Analysis, Algorithms, Applications, Birkhäuser, 1991, pp. 119–123.

[3] K. Cliffe, T. Garratt, A. Spence, Eigenvalues of block matrices arising from problems in fluid mechanics, SIAM Journal on Matrix Analysis and Applications 15 (1994) 1310–1318.

[4] K. Meerbergen, D. Roose, Matrix transformations for computing rightmost eigenvalues of large sparse non-symmetric eigenvalue problems, IMA Journal of Numerical Analysis 16 (1996) 297–346.

[5] D. Sorensen, Implicit application of polynomial filters in a k-step Arnoldi method, SIAM Journal on Matrix Analysis and Applications 13 (1) (1992) 357–385.

[6] R. B. Morgan, On restarting the Arnoldi method for large non-symmetric eigenvalue problems, Mathematics of Computation 65 (215) (1996) 1213–1230.

[7] R. B. Lehoucq, D. C. Sorensen, C. Yang, ARPACK Users Guide: Solution of Large Scale Eigenvalue Problems with Implicitly Restarted Arnoldi Methods, SIAM, Philadelphia, Pennsylvania, USA, 1998.

[8] C. Beattie, M. Embree, J. Rossi, Convergence of restarted Krylov subspaces to invariant subspaces, SIAM Journal on Matrix Analysis and Applications 25 (4) (2004) 1074–1109.
[9] K. Meerbergen, A. Spence, Shift-and-invert iteration for purely imaginary eigenvalues with application to the detection of Hopf bifurcations in large scale problems, SIAM Journal on Matrix Analysis and Applications 31 (4) (2010) 1463–1482.
[10] K. Meerbergen, A. Spence, Implicitly restarted Arnoldi with purification for the shift-invert transformation, Mathematics of Computation 66 (1997) 667–689.
[11] R. Lehoucq, On the convergence of an implicitly restarted Arnoldi method, SIAM Journal on Matrix Analysis and Applications 23 (1999) 551–562.
[12] I. C. F. Ipsen, Computing an eigenvector with inverse iteration, SIAM Review 39 (2) (1997) 254–291.
[13] D. S. Watkins, The Matrix Eigenvalue Problem: GR and Krylov Subspace Methods, SIAM, Philadelphia, Pennsylvania, USA, 2007.
[14] W. Govaerts, Numerical methods for bifurcations of dynamical equilibria, SIAM, Philadelphia, Pennsylvania, USA, 2000.
[15] Y. Saad, Numerical solution of large Lyapunov equations, in: M. A. Kaashoek, J. V. Schuppen, A. C. M. Ran (Eds.), Signal Processing, Scattering, Operator Theory and Numerical Methods, Birkhäuser, Boston, MA, 1990, pp. 503–511.
[16] M. Robbé, M. Sadkane, Use of near-breakdowns in the block Arnoldi method for solving large Sylvester equations, Applied Numerical Mathematics 58 (4) (2008) 486–498.
[17] V. Simoncini, A new iterative method for solving large-scale Lyapunov matrix equations, SIAM Journal on Scientific Computing 29 (3) (2008) 1268–1288.
[18] V. Simoncini, V. Druskin, Convergence analysis of projection methods for the numerical solution of large Lyapunov equations, SIAM Journal on Numerical Analysis 47 (2) (2009) 828–843.
[19] I. M. Jaimoukha, E. M. Kasenally, Krylov subspace methods for solving large Lyapunov equations, SIAM Journal on Numerical Analysis 31 (1) (1994) 227–251.
[20] D. Y. Hu, L. Reichel, Krylov-subspace methods for the Sylvester equation, Linear Algebra and its Applications 172 (1992) 283–313.
[21] A. S. Hodel, B. Tenison, K. R. Poolla, Numerical solution of the Lyapunov equation by approximate power iteration, Linear Algebra and its Applications 236 (1996) 205–230.
[22] T. Gudmundsson, A. J. Laub, Approximate solution of large sparse Lyapunov equations, IEEE Transactions on Automatic Control 39 (5) (1994) 1110–1114.
[23] D. Sorensen, Y. Zhou, Approximate implicit subspace iteration with alternating directions for lti system model reduction, Numerical Linear Algebra with Applications 15 (9) (2008) 873–886.
[24] R. Nong, D. C. Sorensen, A parameter free ADI-like method for the numerical solution of large scale Lyapunov equations, Tech. Rep. TR09-16, Rice University, CAAM, Houston, Texas (2009).
[25] R. Smith, Matrix equation xA+ Bx= C, SIAM Journal on Applied Mathematics 16 (1) (1968) 198–201.
[26] S. Gugercin, J. Li, Smith-type methods for balanced truncation of large sparse systems, Springer Verlag, Berlin, Heidelberg, 2005.
[27] V. Simoncini, Extended Krylov subspace for parameter dependent systems, Applied Numerical Mathematics 60 (5) (2010) 550–560.
[28] T. Penzl, Eigenvalue decay bounds for solutions of Lyapunov equations: the symmetric case, Systems and Control Letters 40 (2000) 139–144.
[29] A. C. Antoulas, D. C. Sorensen, Y. Zhou, On the decay rate of Hankel singular values and related issues, Systems & Control Letters 46 (2002) 323–342.
[30] R. A. Horn, C. R. Johnson, Topics in Matrix Analysis, Cambridge University Press, Cambridge, 1991.
[31] R. H. Bartels, G. W. Stewart, Solution of the matrix equation AX + XB = C, Communications of the ACM 15 (9) (1972) 820–826.
[32] Y. Saad, Numerical Methods for Large Eigenvalue Problems, Manchester University Press, Manchester, United Kingdom, 1992.
[33] D. C. Sorensen, Numerical methods for large eigenvalue problems, Acta Numerica 11 (2002) 519–584.
[34] A. B. J. Kuijlaars, Which eigenvalues are found by the Lanczos method?, SIAM Journal on Matrix Analysis and Applications 22 (1) (2000) 306–321.
[35] B. Nour-Omid, B. Parlett, T. Ericsson, P. Jensen, How to implement the spectral transformation, Mathematics of Computation 48 (1987) 663–673.
[36] W. Olmstead, W. Davis, S. Rosenblat, W. Kath, Bifurcation with memory, SIAM Journal on Applied Mathematics 46 (2) (1986) 171–188.
[37] M. Grinfeld, Dynamics of a model equation in viscoelasticity, in: Proceedings Equadiff '91, Vol. II, World Scientific, River Edge, NJ, 1993, pp. 568–572.
[38] J. Turner, Buoyancy effects in fluids, Cambridge University Press, Cambridge, UK, 1973.
[39] K. Cliffe, T. Garratt, A. Spence, Eigenvalues of the discretized Navier-Stokes equation with application to the detection of Hopf bifurcations, Advances in Computational Mathematics 1 (1993) 337–356.